CLAIMS:

What is claimed is:

1.A method for asynchronous execution within a program, comprising:

executing code in a first thread;

determining whether a first keyword exists in the code, the first keyword indicating a code element that may be executed out of order; and

executing the code element in a second thread.

2.The method of claim 1, wherein the code element is one of an instruction, a block, and a method.

3.The method of claim 1, wherein the first keyword exists in a definition of a method.

4.The method of claim 1, wherein the first thread is executed on a first processor and the second thread is executed on a second processor.

5.The method of claim 1, further comprising:

determining whether a second keyword exists in the code, the second keyword indicating that execution of the code element in the second thread must complete before the next code element is executed; and

executing the next code element in the first thread after execution of the code element in the second thread completes.

6.The method of claim 1, further comprising:

determining whether a third keyword exists in the code element, the third keyword indicating a statement that may be executed out of order; and

executing the statement in a third thread.

7.The method of claim 1, wherein the method is executed by an interpreter.

8.The method of claim 7, wherein the interpreter is a Java virtual machine.

9.The method of claim 1, wherein the second thread is a light weight thread.

10.An apparatus for asynchronous execution within a program, comprising:

first execution means for executing code in a first thread;

determination means for determining whether a first keyword exists in the code, the first keyword indicating a code element that may be executed out of order; and

second execution means for executing the code element in a second thread.

11.The apparatus of claim 10, wherein the code element is one of an instruction, a block, and a method.

12.The apparatus of claim 10, wherein the first keyword exists in a definition of a method.

13.The apparatus of claim 10, wherein the first thread is

executed on a first processor and the second thread is executed on a second processor.

14.The apparatus of claim 10, further comprising:

means for determining whether a second keyword exists in the code, the second keyword indicating that execution of the code element in the second thread must complete before the next code element is executed; and

means for executing the next code element in the first thread after execution of the code element in the second thread completes.

15.The apparatus of claim 10, further comprising:

means for determining whether a third keyword exists in the code element, the third keyword indicating a statement that may be executed out of order; and

means for executing the statement in a third thread.

16.The apparatus of claim 10, wherein the second thread is a light weight thread.

17.An apparatus for asynchronous execution within a program, comprising:

an interpreter; and

a program, the program including a first keyword indicating a code element that may be executed out of order,

wherein the interpreter, upon detecting the keyword, creates a light weight thread and executes the code element in the light weight thread.

18.The apparatus of claim 17, wherein the interpreter is a Java virtual machine.

19.A computer program product, in a computer readable medium, for asynchronous execution within a program, comprising:

instructions for executing code in a first thread;

instructions for determining whether a first keyword exists in the code, the first keyword indicating a code element that may be executed out of order; and

instructions for executing the code element in a second thread.

20.The computer program product of claim 19, wherein the first thread is executed on a first processor and the second thread is executed on a second processor.

21.The computer program product of claim 19, further comprising:

instructions for determining whether a second keyword exists in the code, the second keyword indicating that execution of the code element in the second thread must complete before the next code element is executed; and

instructions for executing the next code element in the first thread after execution of the code element in the second thread completes.

22.The computer program product of claim 19, further comprising:

instructions for determining whether a third keyword

exists in the code element, the third keyword indicating a statement that may be executed out of order; and

instructions for executing the statement in a third thread.